**Input range has label**
Specifies whether the first row or column of the input array is a label and not part of the data to be analyzed.

**Columns / Rows**
Specifies whether the data to be analyzed is organized in columns or rows.

**Inverse**
If ticked, the tool calculates an inverse Discrete Fourier Transform.

**Output in polar form**
If ticked, the tool outputs the results in polar coordinates (that is, magnitude and phase).

**Minimum magnitude for polar form output**
This option is only relevant when you select to output results in polar form. All frequency components with magnitude less than the specified value in decibels will be suppressed with a zero magnitude-phase entry. This is useful when looking at the magnitude-phase spectrum of a signal because there is always some tiny rounding error when performing FFT algorithms, which results in incorrect non-zero phase for non-existent frequencies. By providing a suitable value to this parameter, these non-existent frequency components can be suppressed.

**Tip**

Use the **Shrink** / **Expand** buttons next to the *Input range* and *Results to* fields if you need to shrink the dialog while selecting cells with the mouse.

**Tip**

Calc utilizes the small, otherwise blank area above the **Help**, **OK**, and **Cancel** buttons to provide feedback on erroneous selections on the dialog. For example, the text “Output address is not valid.” appears if you have not entered a valid cell range in the *Results to* field, and in this circumstance the **OK** button is grayed.

To provide an example of using this tool, we make use of the input data set shown in columns B (real values) and C (imaginary values) of the spreadsheet shown in Figure 360. The data shown in columns E (real values) and F (imaginary values) of the spreadsheet are the Fourier transform results calculated by the tool for this input data, using the settings shown in Figure 359.

**Note**

For those with a technical interest in the algorithms used by the Fourier Analysis tool, a radix-2 decimation-in-time FFT is used when the length of the input sequence is an even power of 2, while Bluestein’s FFT algorithm is used when the length of the input sequence is not an even power of 2.

**Tip**

For more information on Fourier analysis, refer to the corresponding Wikipedia article at https://en.wikipedia.org/wiki/Fourier_analysis.